**«** **Microsoft Azure DevOps (Session1)»**

**معاونت** توسعه و برند

واحد نرم‌افزار

****

**فهـرســت**

[**Microsoft Azure DevOps What is؟ 2**](#_Toc61855329)

[**Continuous Delivery 3**](#_Toc61855330)

[**The technologies we need for DevOps 3**](#_Toc61855331)

[**The main DevOps modules 3**](#_Toc61855332)

[**How does Azure DevOps interact with other tools? 4**](#_Toc61855333)

[**Azure Boards 4**](#_Toc61855334)

[**Azure Repos 4**](#_Toc61855335)

[**Azure Pipelines 4**](#_Toc61855336)

[**Azure Test Plans 5**](#_Toc61855337)

[**Azure Artifacts 5**](#_Toc61855338)

[**The Punchline 5**](#_Toc61855339)

[**Developer Division 5**](#_Toc61855340)

[**Planning 6**](#_Toc61855341)

[**Code Flow 6**](#_Toc61855342)

[**FeedBack 6**](#_Toc61855343)

[**Why Change؟ 6**](#_Toc61855344)

[**Sprint Planning 3 Questions 7**](#_Toc61855345)

[**Roles 7**](#_Toc61855346)

[**Teams 7**](#_Toc61855347)

[**Check the tool itself 8**](#_Toc61855348)

[**Technical Dept 9**](#_Toc61855349)

[**Rules 10**](#_Toc61855350)

[**Check Team Team details 10**](#_Toc61855351)

[**Area 10**](#_Toc61855352)

[**Dashboard 11**](#_Toc61855353)

[**Wiki 11**](#_Toc61855354)



Microsoft Azure DevOps What is؟

ترکیبی از افراد، فرآیندها و محصولات است. معمولا در ایران Peaple و Process را در نظر نمی‌گیرند و بر روی یکسری Products مثل Azure DevOps، Atlasian، Gitlab و ... تمرکز می‌کنند در صورتی که اصل کار روی Process و Peaple است (برنامه‌ریزی برای محصولات و ...).

برای ایجاد تغییری که اثر آن در سازمان بماند باید تغییر تدریجی ایجاد نماییم. اگر به عنوان مثال پروژه‌ای تعریف کردیم که در یک سال، کاری را انجام دهیم (Big Bang) نتیجه‌ای دریافت نمی‌کنیم.



## Continuous Delivery

* Build & Test
* Deploy
* Operate
* Monitor & Learn
* Plan & Track
  + برنامه ریزی برای محصولات، مثلا بازنشسته کردن یک محصولی که مشتری ندارد و ...
* Develop

 The technologies we need for DevOps

1. Continuous integration (CI)
2. Continuous Deployment (CD)
3. Learning & Monitoring Continuous



The main DevOps modules

1. Azure Board ( انجام مدیریت محصول و Task‌ها و تست‌های Manuall )
2. Azure Pipelines (همان بحث CI/CD است. یعنی صرفه‌جویی در زمان برای Update کردن IoS یا Web Server و ..)
3. Azure Repos ( با Azure Boards، ارتباط مستقیمی دارد. در حقیقت محل نگهداری کدها می‌باشد. )
4. Azure Test Plans ( در انجام تست‌هایی مثل Test Manuall، Test Automated و ... به ما کمک می‌کند. )
5. Azure Artifacts ( به ما کمک می‌کند تا Package‌هایی که با آن‌ها کار می‌کنیم را Cash نماییم. )



How does Azure DevOps interact with other tools?

ابزارهایی مثل Azure DevOps وظیفه آن‌ها ایجاد خط تولید نرم‌افزار برای سازمان است. در حقیقت این ابزار سعی می‌کند با سایر نرم‌افزارها و زبان‌های برنامه‌نویسی دیگر از قبیل C# و VB.NET تعامل برقرار نماید.

**** Azure Boards

به Scrum و Process‌ها و فرایندها اشاره دارد. در ابزار چه فرآیندهای مختلفی داریم که می‌تواند Workflowzهای مختلف را Support کند. چگونه تیم‌های مختلف (UI، UX، Front و ...) را بر روی کارهای مختلف سازمان در تعامل با یکدیگر قرار دهیم.



Azure Repos

در حقیقت نشان دهنده نحوه استفاده درست از این ابزار است تا قسمت های دیگر دچار مشکل نشوند. اگر Azure Repos را درست تنظیم نشود در آینده در پیاده سازی CI/CD دچار مشکل خواهیم شد.



Azure Pipelines

در حقیقت به این معنا می‌باشد که در حیطه‌ای که تکنولوژی اجازه می‌دهد می‌توانیم کدهای Power Shell، Bash و ... را توسط ‌Pipelineها اتوماتیک نماییم.



# Azure Test Plans

در کل بحث Capture کردن تست‌ها می‌باشد. همچنین از طریق آن می‌توانید گزارشی از Bug‌هایی که داشتید تهیه نمایید، و این که این Bug‌ها در چه Level‌ی بودند و ... ( در برنامه ریزی برای محصول جدید مؤثر می‌باشد. )



Azure Artifacts

در اصل نگهداری Package هایی می باشد که تیم شما می تواند داشته باشد.



The Punchline

در این قسمت اشاره ای به داستان Microsoft می‌نماییم. در اصل هدف اصلی داشتن تیم‌هایی است که به خودیه خود بتوانند Feature‌هایی که در حال توسعه آن‌ها هستند را مدیریت نمایند. هم چنین Sprint‌های آن‌ها 3 هفته‌ای بوده و برنامه‌ریزی آن‌ها به صورت دوطرفه می‌باشد.



Developer Division

در اصل هدفی که Microsoft در نظر گرفت این بود که به جای این که مثلا 6 ماه برای تولید یک محصول زمان بگذاریم، می‌توانیم در مدت 3 ماه ارائه‌ای از نسخه اولیه آن محصول داشته باشم، که همان مفهومScrum و Agile می‌باشد.

Planning

مسئله ای که وجود داشت دوساله بودن Planning‌های آنان بود، یعنی دو ماه تلاش می‌کردند که Planning انجام دهند و مابقی زمان صرف Millestone‌ها می‌شد.



Code Flow

یعنی 8 تا 12 هفته را صرف کدنویسی می‌کردند و مابقی زمان را صرف تست و Stable کردن سیستم می‌نمودند و زمانی که سیستم Stable می‌شد دوباره کدنویسی می‌کردند و و مابقی زمان صرف تست و Stable کردن می‌شد. پس منطقی نیست که ما بدون در نظر گرفتن کیفیت کدنویسی نماییم و مدت زمان زیادی را صرف از رفع Bug‌ها و بهم ریختگی‌ها نماییم.



## FeedBack

مسئله دیگر Feed یا بازخوردی است که مشتری یا تیم‌های دیگر پس از برنامه‌ریزی کارها می‌دهند و ما زمان کافی برای اعمال آنها نداریم، و درنتیجه محصولی را ارائه می‌دهیم که مشتری از آن راضی نمی‌باشد.



Why Change؟

به دلایل زیر Microsoft به این نتیجه رسید که روش قبل برای توسعه دادن درست نمی‌باشد:

1. زمان کافی برای رفع نیاز مشتری را نداشتند.
2. کارمندها ناراضی بودند.
3. در نظر داشتند که به سمت Cloud بروند.



Sprint Planning 3 Questions

درواقه وجود این Sprint‌های 3 هفته‌ای باعث افزایش روند Develop می‌شود.



Roles

در حقیقت ساختار تیم‌ها بسیار مهم است. در ایران معمولا 3 نوع تیم وجود دارد:

* تیم test
* تیم develop
* تیم مدیریت

Microsoft در ابتدا از این ساختار پیروی می‌کرد ولی مشاهده کرد نتیجه‌ی مطلوبی را ایجاد نمی‌کند، پس تصمیم گرفتند تا ساختار تیم‌ها را عوض نمایند. در اصل به این نتیجه رسیدند که جدا بودن تیم تست درست نمی‌باشد، پس تیم مهندسی Engineering را تشکیل دادند که هر عضوی از این تیم وظیفه توسعه و تست را به طور همزمان بر عهده داشت. همچنین برای رفع نیاز مشتری تیم دیگری وجود دارد (Program Management) که با مشتری‌ها تعامل و گفتگو می‌نمایند تا نیازشان را رفع کنند.



Teams

بهینه‌ترین روش وجود تیم‌های Scrum است به نحوی که خود آن تیم بتواند تمام کارهای خودش را انجام دهد. این بحث در چیدمان Azure DevOps تاثیر می‌گذارد.



Check the tool itself

* **DefaultCollection**
* Azure سعی می‌کند برای مدیریت سازمان‌ها آنها را در 3 لایه بشکند که اولین لایه Collection است (بالاترین سطح برای جدا کردن کارها از یکدیگر).
* پایین تر از لایه Collection لایه دیگری به Project (Team Project) وجود دارد. در نظر داشته باشید که برای مثلا یک تیم 10 نفره نباید یک Team Project ایجاد نماییم.

Xrm-Freight => board => backlogs => مشاهده تیم‌های اسکرام

* برای حفظ ابزار از بهم‌ریختگی تیم‌های Scrum و Agile را به وجود آورید.
* نباید به ازای هر پروژه‌ای در Collection یک Team Project ایجاد نماییم و همچنین باید ساختار تیم‌های اسکرام را حفظ نماییم. چنان چه تعداد Team Project‌ها زیاد شود، نگهداری آنها مشکل می‌شود. پس رویکرد داشتن یکTeam Project می‌باشد.

Collection Setting => بالاترین سطح تنظیماتی است که می‌توانیم داشته باشیم

Basic (default)

ابزار Azure DevOps 4 فرآیند برای استفاده از ابزار

Scrum

Agilee

Process

در اختیار ما قرار می‌دهد.

CMMI

* اگر روی Scrum کلیک نماییم Detail زیر را مشاهده خواهیم نمود:
* Bug
* Epic
* Feature
* Impediment
* Product backlog item
* Task
* Level‌ها بسیار حائز اهمیت هستند.
* Epic (کار بلند مدت)
* Feature (کاری که ممکن است دو یا سه ماه طول بکشد)
* Backlog item (کارهایی که در یک Sprint باید انجام شوند)
* Task ( تنها لایه‌ای که زمان و Time در آن دخیل می‌باشد)
* گاهی برای انجام یک پروژه‌ای که در 3 ماه باید به پایان برسد، پس از گذشت دو یا سه هفته متوجه می‌شویم که این پروژه 9 ماه زمان می‌برد، در این موقعیت باید تصمیم بگیریم که پروژه را کنسل نماییم یا ادامه دهیم که با آن Faild Fast (شکست سریع) می‌گویند.



Technical Dept

درواقع یک Product Owner برای چیدن backlog محصول باید موارد زیر را بداند:

* چه میزان Technical Dept باید در Sprint مورد نظر برطرف شود.
* چه میزان Feature باید Deliver شود.
* چه میزان Bug باید برطرف شود.
* و ...

Technical Dept توانایی این را دارد که یک تیم IT را زمین‌گیر نماید، یک تیم IT را حذف نماید یا محصولی را با شکست مواجه کند.

* شما Collection به Collection می‌توانید فرآیندهای مختلفی داشته باشید، ولی نگهداری آن مشکل می‌شود.



Rules

به وسیله آنها می‌توان یکسری مسائل را محدود نمود. از آن جایی که به مرور زمان تعداد آنها زیاد می‌شود، بهترین الگویی که می‌توان استفاده نمود شروع اسم هر قانون با State می باشد، زیرا هر Rule بر روی یک State قرار دارد. به مثال زیر توجه نمایید:

اگر State work item از نوع New بود فیلد assign to آن اجباری شود.

New-assign to-Required

Check Team Team details

1. به صورت blaid کار می‌کند. برخلاف بعضی نرم‌افزارهای دیگر که تب‌های زیادی را در مرورگر باز می‌کند،Azure DevOps از UI مناسب‌تری برخوردار است و به صورت Blaid کار می‌کند.
2. مفهوم Team در Project Setting

* منظور تیم Scrum است. (تیم‌های که حداکثر تعداد آنها بین هفت تا هشت نفر می‌باشد و در زمینه‌های مختلفی تخصص دارند و از پس کارهایی که به آنها محول می‌نمایید بر می‌آیند.)



Area

مشخص کننده محصولات (به وسیله Area محصولات را می شکنیم)

* درواقع Team Project‌ها از یکدیگر جدا هستند و کسی کارهای Team Project دیگری نمی‌بیند، Repository‌های کد دیگران نمیتواند ببیند یا قادر به مشاهده Pipeline یکدیگر نیستند. بنابراین ساختن چیدمان براساس نیاز سازمان که چه زمانی افراد بتوانند کارهای یکدیگر را مشاهده نمایند و چه زمانی قادر به مشاهده کاهای یکدیگر نباشند، حائز اهمیت است.



Dashboard

فرض کنید از شما تقاضا شده است تا پروژه‌ها را چک نمایید، برای انجام این کار تهیه گزارش از وضعیت پیشرفت پروژه‌ها نیاز نمی‌باشد، فقط کافی است که پروژه Dashboard داشته باشد تا موارد زیر را نشان دهد:

* در Sprint جاری و گذشته چه کارهای انجام شده است.
* در Sprint جاری و گذشته چه کارهایی Deliver شده است.
* در Sprint جاری و گذشته چه تست‌هایی Faild شده است.
* و ...

# Wiki

می‌توانید برای Share اطلاعات مانند Data‌ها، File‌ها، Voice recorder‌ها و ... به جای استفاده از UNC Share از Wiki استفاده نمایید.